Ejercicios de programación

Para cada ejercicio se pide:
- escribir el algoritmo en pseudocódigo
- escribir el programa correspondiente en lenguaje C
- adjuntar el resultado obtenido de la ejecución del programa, con datos propuestos o solicitados en el ejercicio (utilizar redireccionamiento de entrada y salida).

1. Dados 3 valores a, b y c, que corresponden a las longitudes de los lados de un triángulo, determinar si el triángulo es equilátero (3 iguales), isósceles (2 iguales) o escaleno (3 distintos). Además determinar los ángulos en cada vértice (A, B, C) utilizando las siguientes relaciones:

$$\cos A = \frac{c^2 + b^2 - a^2}{2cb}$$

$$sen\ A = \sqrt{1 - \cos^2 A}$$

$$A = arctg(tg\ A) \cdot \frac{180}{\pi} = arctg\left(\frac{sen\ A}{\cos A}\right) \cdot \frac{180}{\pi}$$

   Usar como **datos** obligatorios los siguientes:
   9.0 8.0 7.0
   4.0 4.0 5.0
   77.3644 51.3179 51.377
   3.0 3.0 3.0
   3.0 7.0 9.0
   3.0 4.0 5.0

2. Dados un indicador y un radio, si indicador = 1 calcular el perímetro de la circunferencia ($2\pi r$), si indicador = 2 la superficie del círculo (πr^2), si indicador = 3 la superficie de la esfera ($4\pi r^2$) y si indicador = 4 el volumen de la esfera ($4/3\ \pi r^3$).

3. Hallar el máximo común divisor de un par de números naturales, mediante el algoritmo de Euclides (ver apuntes de algoritmos).

   Usar como **datos** obligatorios los siguientes:
   12 4
   2322 347
   274 38
   13 7

4. Para una lista de N números (reales) calcular la suma, el producto, el promedio, el máximo, el mínimo y la cantidad de datos que son mayores que el promedio (buscar un algoritmo eficiente que en con pocos recorridos realice tales tareas).

5. Dado un vector U de N componentes calcular su módulo $\sqrt{\sum_{1}^{N} u_i^2}$

6. Dado un vector U de N componentes encontrar su versor $\dfrac{u_i}{\sqrt{\sum_1^N u_i^2}} \quad i = 1 \ldots N$

7. Dados dos vectores (A, B) de N componentes calcular su producto escalar y el vector suma.

8. Dadas N coordenadas de puntos en el espacio calcular la cantidad de puntos que se encuentran en cada octante y cuántos sobre los ejes.

9. Una forma de interpolar una serie de N puntos (x_i, y_i) dados como datos es hacer pasar un polinomio por tales puntos. La solución clásica es usar la fórmula de Lagrange que expresada matemáticamente es:

$$p(x) = \sum_{1 \le j \le N} \left(\prod_{\substack{1 \le i \le N \\ i \ne j}} \frac{x - x_i}{x_j - x_i} \right) y_j$$

   Se pide escribir un programa que:
   - lea el número de puntos N, que a su vez es el número de términos del polinomio.
   - lea los puntos a interpolar (x_1, y_1), (x_2, y_2),....,(x_N, y_N).
   - lea el valor x a interpolar.
   - que realice la interpolación según la fórmula anterior.
   - el programa seguirá pidiendo valores a interpolar mediante la pregunta: *Desea continuar (s/n)?*, ante los cual el usuario responderá con el carácter *s* o *n.* En caso afirmativo se pedirá el valor x a interpolar.

   Usar como **datos** obligatorios los siguientes:
   N = 5

X1 = 1.0	Y1 = 2.0
X2 = 4.0	Y2 = 3.0
X3 = 5.0	Y3 = 6.0
X4 = 8.0	Y4 = 7.0
X5 = 9.0	Y5 = 11.0

   X = 15, 2, 5, 7, 8.5

10. Dada una matriz N x N, calcular la suma de los elementos de la diagonal principal, el mayor elemento de cada columna y el mayor elemento de cada fila.

11. Dada una matriz A (m x n), calcular las siguientes normas

$$\|A\|_1 = \max_{1 \le j \le n} \sum_{i=1}^{m} |a_{ij}| \qquad \|A\|_\infty = \max_{1 \le i \le m} \sum_{j=1}^{n} |a_{ij}| \qquad \|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2}$$

12. Dada una matriz, obtener la matriz traspuesta.

13. Programa para calcular el producto matricial de dos matrices (MxN y NxP). Se recuerda que cada elemento $C_{i,j}$ de la matriz resultante se calcula aplicando la siguiente fórmula:

$$C(i,j) = \sum_{k=1}^{N} A(i,k) \cdot B(k,j)$$

Usar como **datos** obligatorios los siguientes:

M = 3   N = 3

1	2	3
1	2	2
1	1	1

P = 2

1	3
2	2
0	1

14. Dado un entero positivo, calcular su factorial. Utilizar como tipo de dato el que más rango ofrece. Cuál es el mayor número que se puede calcular su factorial?

15. Calcular las raíces de la ecuación de segundo grado $A\,x^2 + B\,x + C = 0$, para cualquier valor de los coeficientes A, B y C.

Usar como **datos** obligatorios los siguientes:

A = 4   B = 3   C = 2
A = 1   B = 9   C = 4

16. El polinomio: $P_n(x) = a_0 + a_1 x + a_2 x^2 + ... + a_{n-1} x^{n-1} + a_n x^n$ se representa por los coeficientes **a.** Escribir un programa que calcule $P_n(x)$ en **p** puntos equidistantes del intervalo (x_o,x_1), incluídos ambos extremos. Utilizar el algoritmo de factorización de Horner: $P_n(x)=(\ldots(a_n x+a_{n-1}).x+\ldots+ a_1)*x+ a_0$.

Usar como **datos** obligatorios los siguientes:

X0 = -2   X1 = 2   p = 4
A3 = 3   A2 = 4   A1 = 7   A0 = 9
X0 = -1.5   X1 = 1.5   p = 3
A2 = 12   A1 = 3.2   A0 = 4.31

17. Calcular el valor de la función cos(x) en el intervalo [0,π/2] con un paso de π/16 con un error relativo ε. Utilizar la expresión:

$$\cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \frac{x^8}{8!} ....$$

sabiendo que el error absoluto de la suma es menor que el valor absoluto del último término que se toma.

18. Calcular el valor de la función e^x mediante la fórmula:

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + ...$$

Detener el cálculo cuando $| S_{i+1} - S_i | < 1e^{-6}$ siendo S_i la suma de los primeros i términos de la serie.

Usar como **datos** obligatorios los siguientes:

-5	-6
-4	0
4	8
6	2.7172

19. Un número primo es una cantidad entera que es divisible sólo por sí mismo y por uno. Calcular y escribir una lista con los n primeros números primos. (Sugerencia: Un número será primo si los restos de las divisiones por 2,3… $\sqrt{n}$ son todos no nulos. Comprobar el programa calculando los 500 primeros números primos. Optimizar el algoritmo de forma que cada uno de ellos sólo se calcule una vez.

20. El radio medio geométrico (RMG) de una serie de conductores dispuestos paralelamente (ver figura 1) está definido por la siguiente fórmula:

$$RMG = \sqrt[N^2]{\prod_{k=1}^{N}\prod_{m=1}^{N} D_{km}}$$

Donde: $D_{km} =$ Distancia del conductor k al conductor m, siendo $D_{kk} = r_k = e^{-1/4} \cdot r$ donde r es el radio del conductor k.

Nota: La distancia entre dos puntos ($P_1=(x_1,y_1)$, $P_2=(x_2,y_2)$) es:

$$D_{12} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

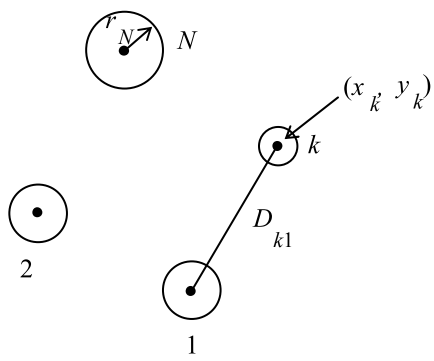


Figura Nº 1

Se pide escribir un programa para calcular el radio medio geométrico (RMG) a partir de la siguiente lectura de datos:

- número de conductores N.
- para cada conductor, leer las coordenadas de sus centros (x_k, y_k) y el radio r_k

El programa seguirá pidiendo otra serie de datos mediante la pregunta *Desea continuar (s/n)?*, ante los cual el usuario responderá con el carácter *s* o *n.*

Nota: para representar los datos de los conductores deben usarse estructuras (struct).

21. Una serie de Fourier expresa una función f(x) definida en el intervalo $-\pi \leq x \leq \pi$ en términos de una serie trigonométrica de la forma:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos nx + b_n sin\, nx \right)$$

Se pide escribir un programa que calcule la serie a partir de los siguientes datos:

- número de componentes n de los vectores. Se considera un valor máximo de 50 componentes.
- los componentes de los vectores a y b, en ese orden.
- valor x para el cálculo de la serie.

El programa seguirá pidiendo valores a calcular mediante la pregunta *Desea continuar (s/n)?*, ante los cual el usuario responderá con el carácter *s* o *n.* En caso afirmativo se pedirá otro valor de x con el que se revaluará el valor de la serie.

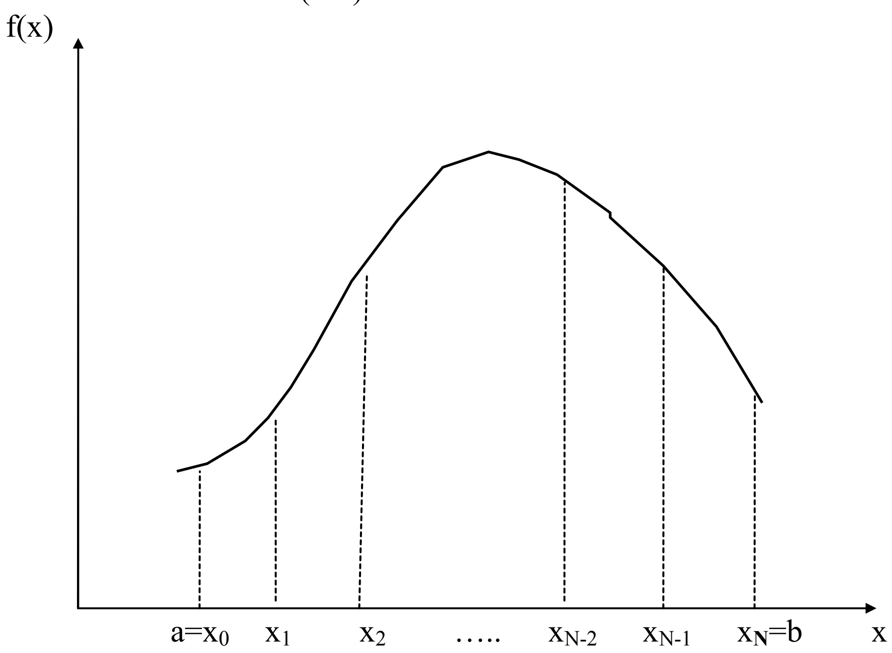
Usar como **datos** obligatorios los siguientes:
n = 5
a0 = 2 a1 = 1 a2 = 0.5 a3 = 0.25 a4 = 0.1 a5 = 0.05
b1 = 2 b2 = 1 b3 = 0.5 b4 = 0.25 b5 = 0.1
x = 1, 0.5, 2, 1.5, 0.25

22. Escribir un programa que realice la integral de una función según el método de Simpson que se describe a continuación:

Parámetros de entrada: intervalo de integración **a, b** y el número de intervalos **N.**
Expresión a aplicar:

$$\int_a^b f(x) \approx \frac{h}{3}\left(f(a) + f(b) + 4 \cdot \sum_{\substack{i=2 \\ Paso\,2}}^{N} f(x_{i-1}) + 2 \cdot \sum_{\substack{i=2 \\ Paso\,2}}^{N} f(x_i) \right)$$

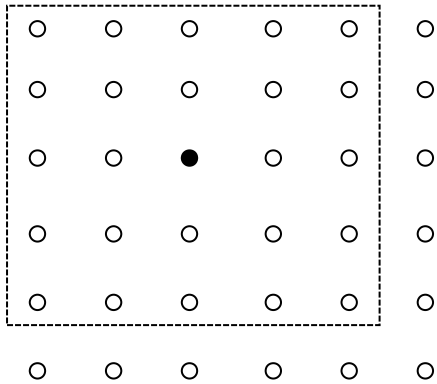
donde: $h=(b-a)/N$



Usar como **datos** obligatorios los siguientes:
f(x) = cos(x)
a = 0 b = 180º
N = 100

23. Una **imagen** (tipo mapa de bits) se considera como una matriz de NxM pixels.Un **pixel** tiene asignado un nivel de gris, que es un número entero sin signo en el rango de 0 a 255 (escala de grises).Una operación para eliminar el ruido en una imagen es la de **suavizado**, que consiste en asignar el valor de intensidad de cada pixel como el valor medio de la intensidad de los pixels en un entorno del pixel (**p**) incluido él mismo (ver figura)

n=2*p + 1



Para entornos cuadrados la media resulta de aplicar:

$$f'(x,y)=\frac{1}{n^2}\sum_{i=-p}^{p}\sum_{j=-p}^{p} f(x+i,y+j)$$

donde: n = 2*p + 1 (entorno de nxn pixels)
p = 1, 2, 3, …

En los bordes hay que asegurar que los índices de los pixels de la imagen estén en el rango correcto y dividir sólo por el número de pixels que cumplen tal condición.
Se pide escribir un programa que reciba como datos:
- Una matriz NxM de pixels
- Un valor entero p, entre 1 y 10, correspondiente al entorno de pixels

Calcule:
- Otra matriz cuyos elementos sean el resultado de la operación de suavizado.

Usar como **datos** obligatorios los siguientes:
N = 5 M = 5 p = 1
1 2 3 4 5
2 3 4 5 6
3 4 5 6 7
1 2 3 4 5
3 4 5 6 7

24. Otra operación común en el tratamiento de imágenes es la obtención del **histograma** de intensidades de los pixels, consistente en hallar, por cada valor posible de la intensidad (0 - 255), el número de pixels que tienen tal valor.Se pide:
- Escribir un programa que reciba como dato una imagen (matriz NxM de pixels) e imprima el histograma de la imagen según el formato:
Nivel de Gris Número de pixels (una fila por cada nivel de gris)

Usar como **datos** obligatorios los siguientes:
N = 5 M = 5
1 2 3 4 5
2 3 4 5 6
3 4 5 6 7
1 2 3 4 5
3 4 5 6 7

25. En el cálculo del impuesto a la renta (Renta 97) es necesario determinar la **cuota íntegra del impuesto** que se calcula sumando la cuota estatal y la cuota autonómica.

Las escalas de gravamen aplicables para la declaración individual son:

	Cuota estatal		Cuota autonómica	
Base liquidable hasta pesetas	Cuota integra (pesetas)	Tipo aplicable al resto (%)	Cuota integra (pesetas)	Tipo aplicable al resto (%)
442.000	0	17,00	0	3,00
1.136.000	117.980	19,55	20.820	3,45
2.305.000	346.520	23,80	61.151	4,20
3.474.000	624.742	27,20	110.249	4,80
4.643.000	942.710	30,60	166.361	5,40
5.812.000	1.300.424	34,00	229.487	6,00
6.981.000	1.697.884	38,25	299.627	6,75
8.150.000	2.145.026	41,65	378.534	7,35
9.319.000	2.631.915	45,05	464.456	7,95
10.488.000	3.158.549	47,60	557.391	8,40

Ejemplo de aplicación: Para una base liquidable de 3.128.362 se determina las cuotas como sigue:

<u>Parte estatal:</u>

Base liquidable	3.128.362			
Hasta (según tabla)	2.305.000			346.520
Resto	823.362	al	23,80 %	195.960
Cuota resultante a:				542.480

<u>Parte autonómica:</u>

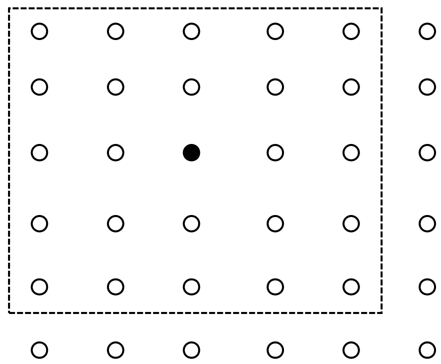
Base liquidable	3.128.362			
Hasta (según tabla)	2.305.000			61.151
Resto	823.362	al	4,20 %	34.581
Cuota resultante b:				95.732

<u>Cuota íntegra del impuesto (a+b):</u> 638.212

Se pide escribir un programa que lea una lista de Bases Liquidables y que calcule la cuota íntegra del impuesto para cada valor imprimiendo tanto la Base Liquidable como la Cuota Integra

26. Una **imagen** (tipo mapa de bits) se representa como una matriz de NxM pixels.Un **pixel** tiene asignado un nivel de gris, que es un número entero (sin signo) en el rango de 0 a 255 (escala de grises). Una operación para eliminar el ruido en una imagen es la de **suavizado mediante mediana**, que consiste en asignar el valor de intensidad de gris de cada pixel como el valor de la mediana de los pixels del entorno (**p**) incluido él mismo (ver figura).
En los bordes hay que asegurar que los índices de los pixels de la imagen estén en el rango correcto y considerar sólo los pixels vecinos que cumplen tal condición.
La **mediana** de una serie de valores es el valor central de los valores ordenados de mayor a menor (Ejm: Caso impar→ La mediana de 2, 4, 5, 8, 10 es 5;
Caso par→ La mediana de 2, 4, 6, 8, 10, 13 es 6).
Nótese, además, que sólo es necesario ordenar la serie de valores hasta el índice que señala la mediana.

n=2*p + 1



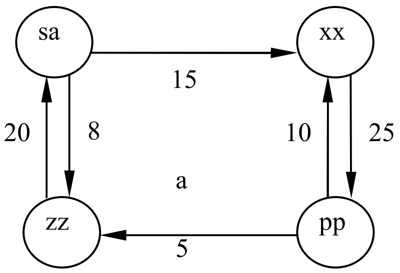
donde: n = 2*p + 1 (entorno de *n*x*n* pixels)
p = 1, 2, 3, …

Se pide:
Escribir un programa que:
- Lea una imagen (matriz de NxM).
- Suavice la imagen anterior mediante mediana, aceptando un valor entero p, entre 1 y 10, correspondiente al entorno de pixels.
- Imprima la imagen suavizada.

Usar como **datos** obligatorios los siguientes:
N = 5 M = 5 p = 1
1 2 3 4 5
2 3 4 5 6
3 4 5 6 7
1 2 3 4 5
3 4 5 6 7

27. En una red telefónica es necesario **calcular** el **tráfico medio ponderado por la distancia** (tmp) entre las centrales de conmutación. La información disponible para cada central es su nombre, y coordenadas geográficas *x, y*. También se dispone del tráfico entre centrales según la matriz de adyacencia entre ellas.

Un **ejemplo de red** se muestra a continuación:



	sa	xx	pp	zz
sa	0	15	0	8
xx	0	0	25	0
pp	0	10	0	5
zz	20	0	0	0

datos "topologia.txt":

```
4
sa   5.0    10.5
xx   15.0   8.0
pp   25.0   18
zz   8.0    22
0    15     0     8
0    0      25    0
0    10     0     5
20   0      0     0
```

La distancia entre dos puntos ($P_1=(x_1,y_1)$, $P_2=(x_2,y_2)$) es:

$$D_{12} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

El tráfico medio ponderado por la distancia (tmp) viene dado por:

$$tmp_i = \frac{\sum_{\substack{j=1 \\ j\neq i}}^{n} t_{ij} d_{ij}}{\sum_{\substack{j=1 \\ j\neq i}}^{n} d_{ij}}$$

donde: tmp_i = tráfico medio ponderado por la distancia de la central *i*
t_{ij} = tráfico de la central *i* a *j*
d_{ij} = distancia de la central *i* a *j*

Se pide elaborar de manera estructurada un programa que **calcule** el tráfico medio ponderado por la distancia (tmp_i) para cada central de conmutación de una red (Máximo 50) y las **imprima de menor a mayor** junto con el nombre de la central, a partir de la lectura de los siguientes datos contenidos:
- Número de centrales de conmutación N
- Para cada central: Nombre (máximo 20 caracteres), coordenadas de posición (x_k, y_k)
- Matriz de configuración de la red: Por cada central se especifica la adyacencia con las demás colocando el número de llamadas, si existe un enlace, y cero (0) si no lo hay.

Nota: usar obligatoriamente estructuras (**struct**) para la especificación de la posición.

Usar como **datos** obligatorios los mencionados en topología.dat

28. Desarrollar un programa que lea dos números complejos (forma cartesiana) y realice las operaciones de suma, resta, multiplicación y división. Además para cada número se convertirá a su forma polar. Para representar los números complejos se deberá usar estructuras (struct)